# Mini Proyecto Asignatura “Tópicos De Seguridad”

Agustín Arias y Johan Rodríguez

Contenido

[Mini Proyecto Asignatura “Tópicos De Seguridad” 1](#_Toc201764795)

[Repositorio 1](#_Toc201764796)

[Contexto 1](#_Toc201764797)

[Solución 1](#_Toc201764798)

[Arquitectura 3](#_Toc201764799)

[Vistas 9](#_Toc201764800)

## Repositorio

Github: https://github.com/iMistDev/TSI.git

## Contexto

Para el caso del banco MGTI, Se nos solicita hacer un login o inicio de sesión con acceso mediante al credencial del usuario, además, esta credencial debe estar vinculada con una solicitud MFA, y para implementar la solicitud de Autenticación Mutlifactor (a partir de ahora MFA) Se nos indica utilizar las aplicaciones de Autenticación como las de Google o Microsoft, y debe haber una conexión con el dispositivo móvil (Smartphone)

## Solución

Para dar solución al caso otorgado, nosotros optamos por levantar una aplicación web en el framework de Python Django, dando el siguiente resultado a nivel de organización de Proyecto como se muestra en la imagen a continuación.

Pantalla de computadora con letras

El contenido generado por IA puede ser incorrecto.

En la imagen antes vista se demuestra la arquitectura del proyecto siendo el nombre de este “SeguridadMGTI”, dentro del proyecto Django, hay una aplicación llamada AuthMGTI la cual será la encargada de gestionar las solicitudes MFA.

Dentro de la Aplicación, los archivos models.py, urls.py y views.py son los encargados de gestionar las solicitudes de ingreso a la aplicación y la autenticación MFA, revisaremos los 3 archivos y la configuración del proyecto en general.

## Arquitectura

Models.py

Texto

El contenido generado por IA puede ser incorrecto.

El archivo contiene una clase simple para registrar la clave OTP (One Time Password) y asociar dicha clave a un usuario

Urls.py

Captura de pantalla con la imagen de una pantalla

El contenido generado por IA puede ser incorrecto.

En este archivo se maneja el enrutamiento de la aplicación que redirigirá a distintas secciones controladas por las vistas o el archivo views.py

Views.py

Texto

El contenido generado por IA puede ser incorrecto.

La primera parte de este archivo son las librerías necesarias para el manejo de la aplicación de seguridad y la conexión con la aplicación externa de autenticación, las segundas librerías son nativas del framework Django y son necesarias para el manejo de la seguridad de la app y validar los inicios de sesión posteriores.

Texto

El contenido generado por IA puede ser incorrecto.

Esta parte del archivo es la función necesaria para hacer un registro de un usuario nuevo dentro de la aplicación, donde se le pide un usuario y contraseña, además valida que no haya redundancias dentro de la base de datos, finalmente genera un objeto de tipo usuario al cual se le asigna una TOTP que es una clave de una uso basada en tiempo con una codificación en base a 32 bits.

Texto

El contenido generado por IA puede ser incorrecto.

Esta función que es la vista de inicio de sesión lo que hace es hacer una pre-validación y un pre-inicio de sesión para el usuario que luego será redirigido a una vista de validación por MFA.

La función recoge de un método POST, el usuario y la contraseña, luego si el usuario es uno existente lo pone una pre-sesión ‘pre\_2fa\_user’ de la cual recoge el id y lo redirige a la verificación para completar el inicio de sesión.

Texto

El contenido generado por IA puede ser incorrecto.

Esta función de las vistas es la más crucial, ya que es la que se encarga de la verificación por MFA, la función parte recogiendo el id del usuario, y luego toma un objeto del tipo usuario que de sea verificar, y si no tiene una clave OTP generada previamente lo hace al momento, finalmente recoge la clave ‘otp secret’ y tanto el nombre de usuario como el nombre de la aplicación que serán pasadas a la app externa de autenticación, a continuación se crea el QR con lo recogido por ‘otp\_uri’ y se crea de manera momentánea con una codificación en base 64, cuando se dice que es de forma momentánea es que la imagen del QR no queda en memoria, finalmente el usuario envía devuelta el código de la app authenticator de su elección, y lo valida, y de ser así lo reenvía al a vista dashboard que requiere que el usuario esté con un inicio de sesión válido.

Texto

El contenido generado por IA puede ser incorrecto.

Además, por si hubo algún fallo en el registro o inicio de sesión tenemos una función que regenera el código QR para el usuario.

Settings.py

Texto

El contenido generado por IA puede ser incorrecto.

La aplicación debe ser reconocida por el proyecto y debe acutar como Middleware, es decir, es una aplicación que va a estar interceptando las peticiones HTTP que llegue a la proyecto, además de eso se debe cargar las liberarías para que sean reconocidas por el proyecto.

# Vistas

Incio de Sesión

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Registro

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Verificación MFA

Código QR

El contenido generado por IA puede ser incorrecto.

Dashboard o Inicio de la App

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.